BAB 3
ANALISIS DAN PERANCANGAN

3.1. Analisis Masalah

Analisis masalah dalam penelitian ini adalah bagaimana cara menerapkan metode *forward chaining* untuk penentuan status karakter pada *game* petualangan. Analisis masalah, pada tahapan ini merupakan salah satu proses mengidentifikasi masalah-masalah yang berkaitan dengan bagaimana suatu sistem akan melakukan proses peninjauan terhadap kondisi karakter yaitu berupa status karakter berdasarkan penelusuran metode inferensi *forward chaining* dengan penelusuran *tree.*

3.2. Analisis *Game* sejenis

Analisis yang dilakukan merupakan observasi dari *game* sejenis untuk melihat bagaimana penerapan metode *forward chaining* pada suatau *game,* dan juga melihat *gameplay* dari *game* sejenis, karena dengan melihat *gameplay* dari suatu *game* bisa dilihat hal apa saja yang harus diteliti. *Game* yang di analisis adalah *game Chicken Roll* untuk melihat penerapan metode *forward chaining* dan *game The Art Collector* untuk melihat *gameplay*.

3.2.1. Analisis Metode *Forward Chaining* pada *Game Chicken Roll*

Game Chicken Roll merupakan *game* yang bertipe leveling, dimana kekuatan *game* terletak pada tingkat kesulitan yang berbeda pada tiap level. *Game* ini berjalan pada *platform desktop* (*Windows*), *game* ini hanya bersifat *single player*. *Game* ini merupakan implementasi konsep *AI* khususnya metode *forward chaining*, pada dasarnya metode *forward chaining* pada *game* ini dimanfaatkan untuk perpindahan level yaitu menentukan apakah user dapat melanjutkan permainan ke level berikutnya atau tidak dan review sebagai hasil dari bermain *game*. Berikut beberapa rule *forward chaining* yang digunakan *game chicken roll* dapat dilihat pada tabel 3.1 dan 3.2[3].

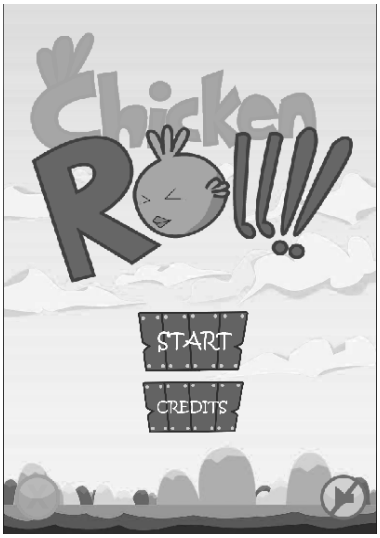
Tabel 3.1 Rule *Game Chicken Roll* Berdasarkan *Review*

No	IF	THEN
1.	R01=’3’ **AND** R02=’G’ **AND** R03=’=>500’ **AND** R04=’M’ **AND** R05=’TT’	H01
	Keterangan	
	Nyawa sisa 3, Target goal, skor => 500, makan, tidak tertabrak.	Congratulations, Piala Emas, Coins 500, Bonus ayam.
2.	R01=’2’ AND R02=’G’ AND R03=’=>500’ AND R04=’TM’ AND R05=’TT’	H02
	Keterangan	
	Nyawa sisa 2, Target goal, skor => 500, tidak makan, tidak tertabrak.	Piala Silver, Coins 500.
3.	R01=’1’ AND R02=’G’ AND R03=’=>500’ AND R04=’TM’ AND R05=’TT’	H03
	Keterangan	
	Nyawa sisa 1, Target goal, skor => 500, tidak makan, tidak tertabrak.	Coins 500.
4.	R01=’0’ AND R02=’TG’ AND R03=’<500’ AND R04=’TM’	H04
	Keterangan	
	Nyawa sisa 0, Target tidak goal, skor < 500, tidak makan.	Game Over.
5.	R01=’0’ AND R02=’TG’ AND R03=’<500’ AND R04=’M’	H05
	Keterangan	
	Nyawa sisa 0, Target tidak goal, skor < 500, makan.	Bonus Ayam.

Tabel 3.2 Rule *Game Chicken Roll* Berdasarkan Level

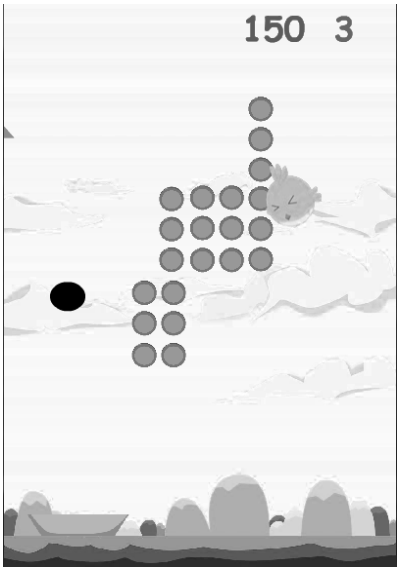
No	IF	THEN
1.	H01	L01
	Keterangan	
	Congratulations, Piala Emas, Coins 500, Bonus ayam.	Lanjut level.
2.	H02	L01
	Keterangan	
	Piala Silver, Coins 500.	Lanjut level.
3.	H03	L01
	Keterangan	
	Coins 500.	Lanjut level.
4.	H04	L02
	Keterangan	
	Game Over.	Ulang level.
5.	H05	L01
	Keterangan	
	Bonus Ayam.	Lanjut level.

Beberapa *screenshot game Chicken Roll* dapat dilihat pada gambar 3.1 sampai 3.3[3].



Gambar 3.1 Tampilan Utama Chicken Roll

Gambar 3.2 menunjukkan level 1 pada *game chicken roll*. Pada *scene* ini, user diminta untuk menyelesaikan permainan pada level 1 ini. Ketika user selesai memainkan level 1 ini maka metode *forward chaining* akan bekerja sesuai dengan rule yang telah di tetapkan.



Gambar 3.2 Tampilan Level 1 *Game Chicken Roll*

Gambar 3.3 berikut ini menunjukkan tampilan review H01 pada *game chicken roll*. Review ini akan muncul ketika user selesai memainkan suatu level dalam *game* sesuai kondisi yang sudah ditentukan. Kondisi yang berbeda akan mendapatkan review yang berbeda pula sesuai dengan rule yang sudah ditetapkan.



Gambar 3.3 Tampilan Review H01 *Game Chicken Roll*

3.2.2. Analisis *Game Play* pada *Game The Art Collector*

Game The Art Collector merupakan *game* bergenre *adventure*, yang dibuat oleh Alastair Low. *Game* ini dibuat menggunakan *Blender Game Engine* (*BGE*) yang mempunyai desain tiga dimensi (*3D*). Pada *game* ini pemain berperan sebagai seorang anak muda yang harus menyelamatkan seorang pria tua yang terjebak dalam sebuah lukisan oleh Villan dari lukisan lain, dan menjelajahi empat lukisan yang unik dengan gaya seni berbeda.

Gameplay dari *game* ini yaitu berupa interaktif antara *player* sebagai karakter dan lingkungan *game* itu sendiri. *Gameplay* dapat dilihat pada tabel 3.3.

Tabel 3.3 *Gameplay The Art Collector*

Gameplay	
1	Karakter berjalan dengan menekan *keyboard* W
2	Karakter mundur dengan menekan *keyboard* S
3	Karakter belok kiri dengan menekan *keyboard* A
4	Karakter belok kanan dengan menekan *keyboard* D
5	Karakter berlari dengan menekan *keyboard* Shift
6	Karakter melompat dengan menekan *keyboard* Space
7	Karakter menyerang dengan *mouse* LMB
8	Karakter melihat dengan pergerakan *mouse*

Beberapa *screenshot* peraminan dari *game The Art Collector* dapat dilihat pada gambar 3.4 sampai 3.6.



Gambar 3.4 Menu Utama *The Art Collector*

Gambar 3.4 merupakan menu utama dari *game The Art Collector*, pada menu utama *game* ini menampilkan nama *game*, menampilkan *icon credits*, menampilkan *icon instuction* dan *icon start*.



Gambar 3.5 Tampilan pada Lukisan Kedua

Gambar 3.5 merupakan tampilan pada lukisan/lokasi kedua pada *game The Art Collector*, pada gambar ini terlihat karakter sedang melompat, haltersebut

merupakan *gameplay* dari *game* ini, yaitu dengan cara menekan keyboard space, maka karakter akan melompat.



Gambar 3.6 Tampilan pada Lukisan Keempat

Gambar 3.6 merupakan tampilan pada lukisan/lokasi keempat pada *game The Art Collector*, pada gambar ini terlihat karakter sedang memegang palet sebagai tameng dan kuas sebagai senjata, dan trelihat pula musuh yang bernama Villan. *Gameplay* pada lokasi ini yaitu karakter menyerang musuh dengan menekan *mouse* LMB.

3.3. Analisis *Game* yang Akan Dikembangkan

Analisis *game* yang akan dikembangkan merupakan bagian yang mendeskripsikan *game* yang akan dikembangkan. Analisis pada ada bagian ini terdiri dari arsitektur, *storyline*, *gameplay*.

3.3.1. Arsitektur

Game ini di bangun dalam bentuk (*3D*), masih berupa simulator, serta fiture yang ditunjukan pada tabel 3.4.

Tabel 3.4 Fitur *Game* Petualangan

Fitur *Game*	
1	Sistem permainan *single player*
2	Pemain berperan sebagai karakter dalam *game*
3	Pemain akan mengalahkan beberapa musuh dalam *game*
4	Terdapat *Head up display* pada *game*
5	Pada karakter terdapat hp,exp,level dan skill

3.3.2. *Storyline*

Storyline merupakan cerita pada *game* dimana pada *storyline game* akan ditentukan bagaimana jalan cerita *game*, dan aturannya. *Game* petualangan ini sebenarnya hanya berupa simulator sebagai bahan implementasi metode *forward chaining* untuk penentuan status karakter. Simulator *game* petualangan ini sama halnya dengan *game* petualangan yang ada selama ini, yaitu menitikberatkan permainan dimana pemain berperan sebagai karakter protagonis dalam sebuah cerita interaktif yang didorong oleh explorasi, menyelesaikan berbagai tantangan dan mengalahkan musuh.

3.3.3. *Gameplay*

Gameplay merupakan suatu deskripsi dari alur permainan, dari mulai awal permainan sampai akhir permainan, berikut tabel *gameplay game* petualangan dapat dilihat pada tabel 3.5.

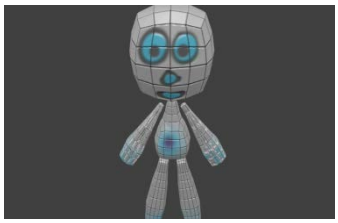
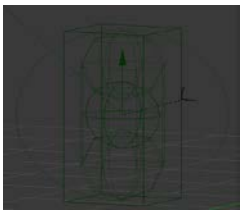
Tabel 3.5 *Gameplay* pada *Game* Petualangan

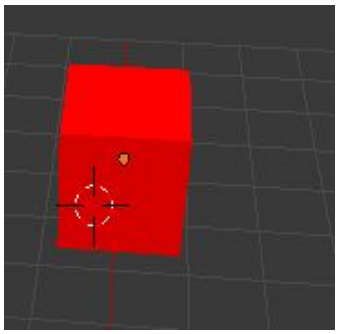
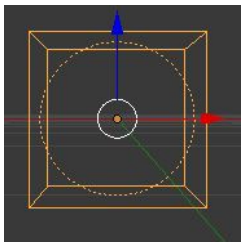
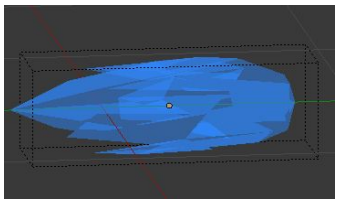
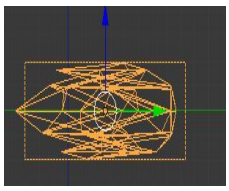
Gameplay	
1	Karakter berjalan dengan menekan *keyboard* W
2	Karakter mundur dengan menekan *keyboard* S
3	Karakter belok kiri dengan menekan *keyboard* A
4	Karakter belok kanan dengan menekan *keyboard* D
5	Karakter berlari dengan menekan *keyboard* Shift
6	Karakter melompat dengan menekan *keyboard* Space
7	Karakter menyerang dengan *keyboard* E, One dan Two
8	Karakter melihat dengan pergerakan *mouse*

3.4. Analisis Masukan

Masukan yaitu berdasarkan *collision* yang terjadi pada karakter dan musuh, *collision* disini berperan sebagai masukan untuk *value* hp dan exp yang nantinya akan digunakan sebagai *rule* pada metode *forward chaining* untuk penentuan status karakter berupa informasi mengenai *fireboll*, *iceboll* dan *gameover*. Area *collision* terdapat pada beberapa konten, diantaranya dapat dilihat pada tabel 3.6.

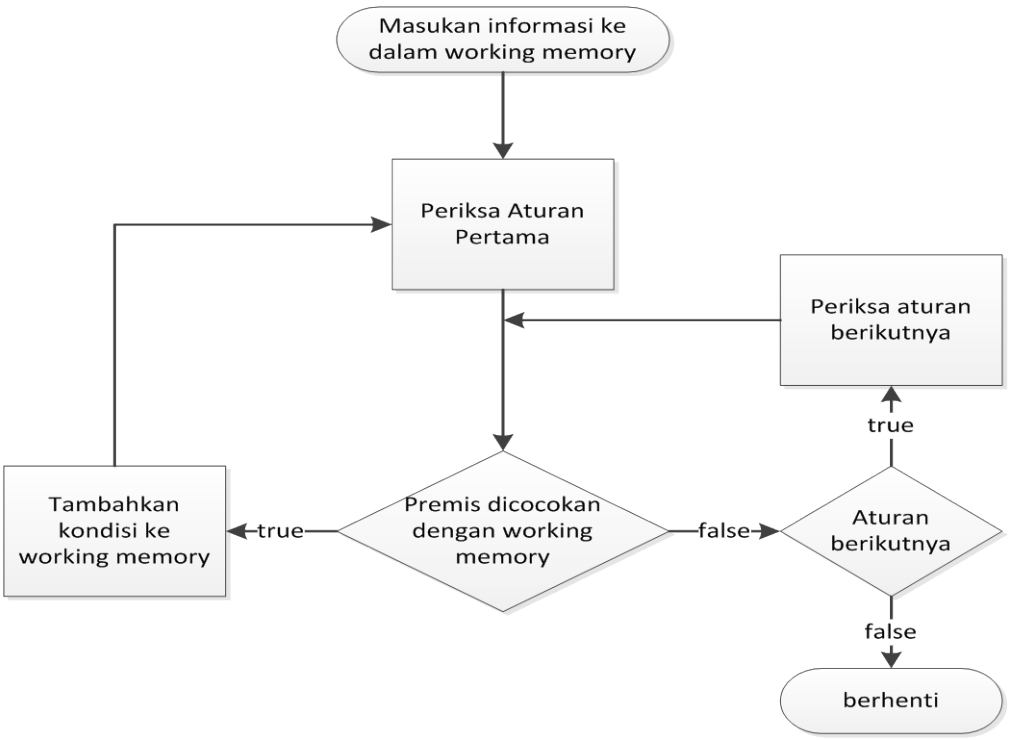
Tabel 3.6 Konten dan Area *Collision*

No	Nama	Gambar	Area Collision	Keterangan
1	Player			Area *collision* pada player terletak didalam player

No	Nama	Gambar	Area Collision	Keterangan
2	Musuh			Area *collision* musuh terletak didalam musuh
3	Iceboll			Area *collision* iceboll terletak diluar iceboll

3.5. Analisis Metode

Metode *forward chaining* adalah metode yang berjalan dari gejala menuju kesimpulan, pertama yang dilakukan adalah membuat tabel premis atau pernyataan yang berisi suatu fakta-fakta yang berpengaruh terhadap proses pengecekan, kedua hal tersebut akan berpengaruh terhadap penentuan status karakter.Alur metode *forward chaining* dapat dilihat pada gambar 3.7.



Gambar 3.7 Skema *Forward Chaining*

Berikut status karakter yang akan ditentukan pada *game* petualangan, dapat dilihat pada tabel 3.7.

Tabel 3.7 Tabel Status

Kode	Status
S01	Add fireboll
S02	Add iceboll
S03	Game Over

Berikut fakta-fakta yang ada pada *game* petualangan untuk menentukan status karakter, dapat dilihat pada tabel 3.8.

Tabel 3.8 Tabel Fakta

Kode	Fakta
F1	Karakter hit musuh
F2	Musuh mati
F3	Musuh tidak mati
F4	Exp karakter bertambah
F5	Musuh hit karakter
F6	Exp=100
F7	Karakter level 1
F8	Get skill 1
F9	Karakter level 2
F10	Use skill 1
F11	Get skill 2
F12	Use skill 2
F13	Hp karakter berkurang
F14	Hp=0

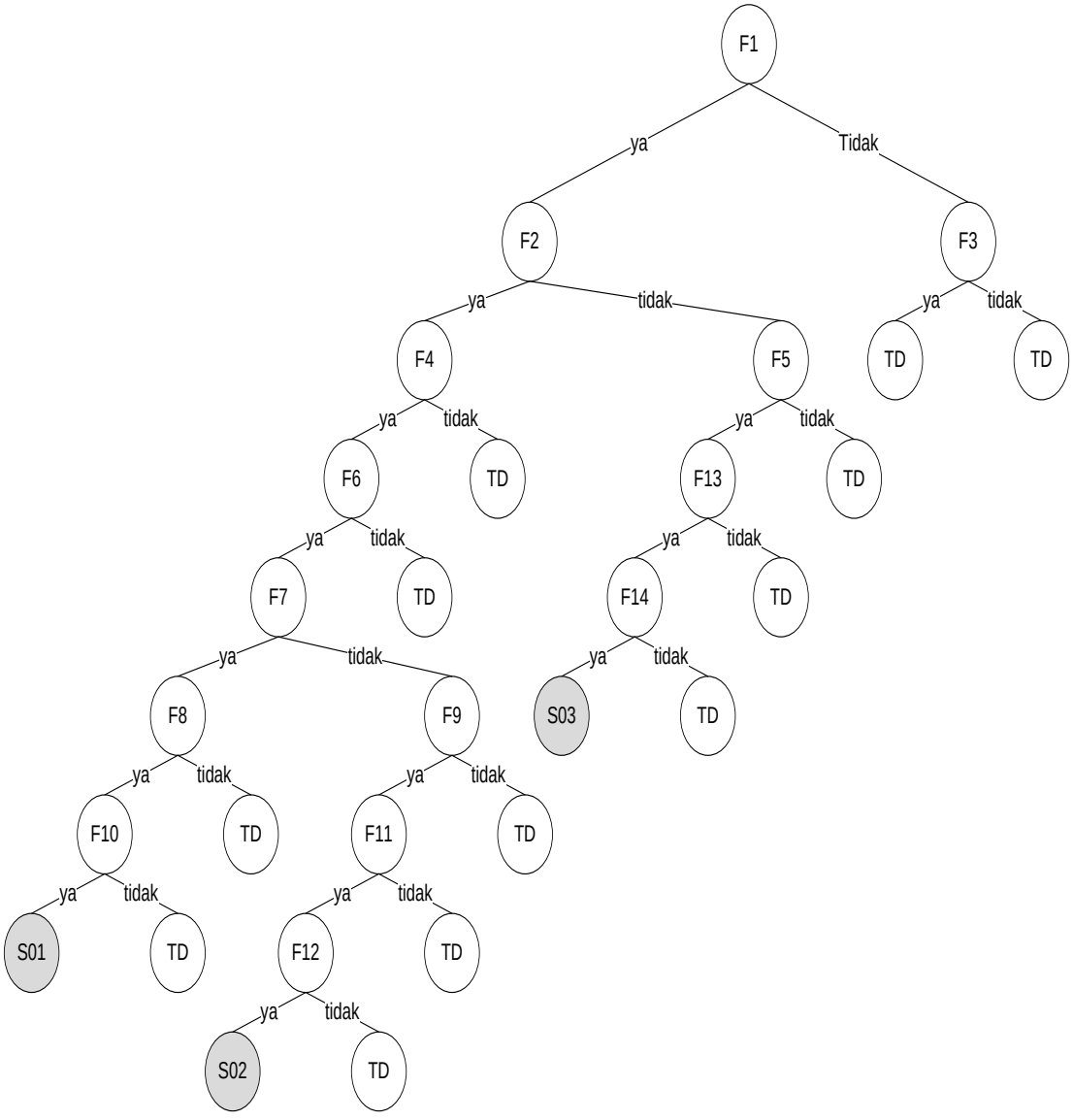
Berikut merupakan *rule forward chaining* untuk penentuan status karakter yang ditunjukan pada tabel 3.9.

Tabel 3.9 Tabel Rule

RULE	FAKTA													
	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14
S01	*	*		*		*	*	*		*				
S02	*	*		*		*			*		*	*		
S03	*				*								*	*

3.5.1. Inferensi *Forward Chaining* Berdasarkan Rule

Inferensi dari *forward chaining* adalah mekanisme berfikir dan pola-pola penelaran yang digunakan sistem untuk mencapai suatu kesimpulan. Inferensi *forward chaining* menggunakan pohon penelusuran bertipe *binary* tidak general dikarenakan jawaban hanya dua, yaitu ya dan tidak. Pohon penelusuran dapat dilihat pada gambar 3.8.



Gambar 3.8 Pohon Penelusuran

Keterangan :

1. F1,F2..F14 : Kode Fakta
2. S01,S02,S03 : Kode Status
3. TD : Tidak Diketahui

Fakta ini merupakan basis pengetahuan untuk membuat suatu kesimpulan yang menjadi *goal*. Keterhubungan/relasi kode fakta dan kode status dapat dilihat pada tabel 3.10.

Tabel 3.10 Status Karakter dan Fakta

Keterangan	Simbol	Status Karakter		
		S01	S02	S03
Karakter hit musuh	**F1**	*	*	*
Musuh mati	**F2**	*	*	*
Musuh tidak mati	**F3**			
Exp karakter bertambah	**F4**	*	*	
Musuh hit karakter	**F5**			*
Exp=100	**F6**	*	*	
Karakter level 1	**F7**	*	*	
Get skill 1	**F8**	*		
Karakter level 2	**F9**		*	
Use skill 1	**F10**	*		
Get skill 2	**F11**		*	
Use skill 2	**F12**		*	
Hp karakter berkurang	**F13**			*
Hp=0	**F14**			*

Keterangan :

1. F1,F2..F14 : Kode Fakta
2. S01,S02,S03 : Kode Status
3. * : Relasi kode fakta dan kode status

3.6. Spesifikasi Kebutuhan Perangkat Lunak

Spesifikasi kebutuhan perangkat lunak dilakukan berdasarkan kebutuhan *game*. Spesifikasi kebutuhan perangkat lunak akan dibagi kedalam dua bagian yaitu SKPL-F (Spesifikasi Kebutuhan Perangkat Lunak Fungsional) dan SKPL-NF (Spesifikasi Kebutuhan Perangkat Lunak Non-Fungsional). Spesifikasi kebutuhan perangkat lunak dalam penelitian ini dapat dilihat pada Tabel 3.11 sampai Tabel 3.13.

Tabel 3.11 SKPL *User Requirement*

Kode	Kebutuhan
SKPL-F1	Perangkat lunak dapat memproses hasil masukan berupa key control untuk pergerakan karakter.
SKPL-F2	Perangkat lunak dapat memproses hasil masukan berupa event atau fakta untuk kemudian dimulai proses penelusuran untuk mengetahui kesimpulan.
SKPL-F3	Perangkat lunak dapat menampilkan hasil proses penelusuran berupa status karakter.

Tabel 3.12 SKPL *System Requirement*

Kode	Kebutuhan
SKPL–F1	Proses masukan yang diterima sistem berupa *collision* yang dihasilkan dari suatu kejadian(*event*) berupa fakta didalam *game* yang kemudian diolah menjadi keluaran yang berupa kesimpulan sesuai dengan *rule.*
SKPL-F2	1. Kesimpulan berupa status karakter yang berada pada HUD. 2. Status karakter terdiri dari : fireboll, iceboll, dan game over.

Tabel 3.13 SKPL Non-Fungsional

Kode	Kategori	Kebutuhan
SKPL-NF1	*Efficiency* *(Product Requirement)*	Sistem dapat memproses masukan *collision*
SKPL-NF2	*Development* *(Organizatonal* *Requirement)*	Untuk menjalankan sistem yang di bangun maka dibutuhkan : a. *Microsoft Windows* sebagai sistem operasi b. *Blender game engine* sebagai alat untuk pengembangan *game*

3.6.1. Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional adalah langkah dimana seorang pembangun perangkat lunak menganalisis sumber daya yang akan digunakan dan menggunakan perangkat lunak yang dibangun. Perangkat keras dan perangkat lunak yang dimiliki harus sesuai dengan kebutuhan, sehingga dapat ditentukan kompabilitas aplikasi yang dibangun terhadap sumber daya yang ada.

3.6.2. Analisis Kebutuhan Perangkat Keras

Perangkat keras yang dibutuhkan untuk implementasi *forward chaining* pada *game* yang akan dibuat dapat dilihat pada Tabel 3.14.

Tabel 3.14 Spesifikasi Kebutuhan Perangkat Keras

NO	Perangkat Keras	Spesifikasi
1	Prosesor	1.80 Ghz
2	Monitor	Monitor 14’’, resolusi 1024x768
3	VGA	Kecepatan 512MB
4	RAM	2GB
5	Keyboard	Standard
6	Mouse	Standard
7	Memori	2GB

3.6.3. Analisis Kebutuhan Perangkat Lunak

Perangkat lunak yang dibutuhkan untuk implementasi forward chaining pada game yang akan dibuat. Adapun perangkat lunak yang dibutuhkan dapat dilihat pada Tabel 3.15.

Tabel 3.15 Spesifikasi Kebutuhan Perangkat Lunak

NO	Perangkat Lunak	Keterangan
1	Sistem Operasi	Windows 7
2	*Tools* Program	Blender3d
3	*Tools* Pengembangan	Blender Game Engine

3.6.4. Analisis Kebutuhan Pengguna

Analisis kebutuhan pengguna digunakan untuk mengetahui spesifikasi pengguna untuk dapat memainkan *game* petualangan yang masih berupa simulator, spesifikasinya dapat dilihat pada Tabel 3.16.

Tabel 3.16 Spesifikasi Kebutuhan Pengguna

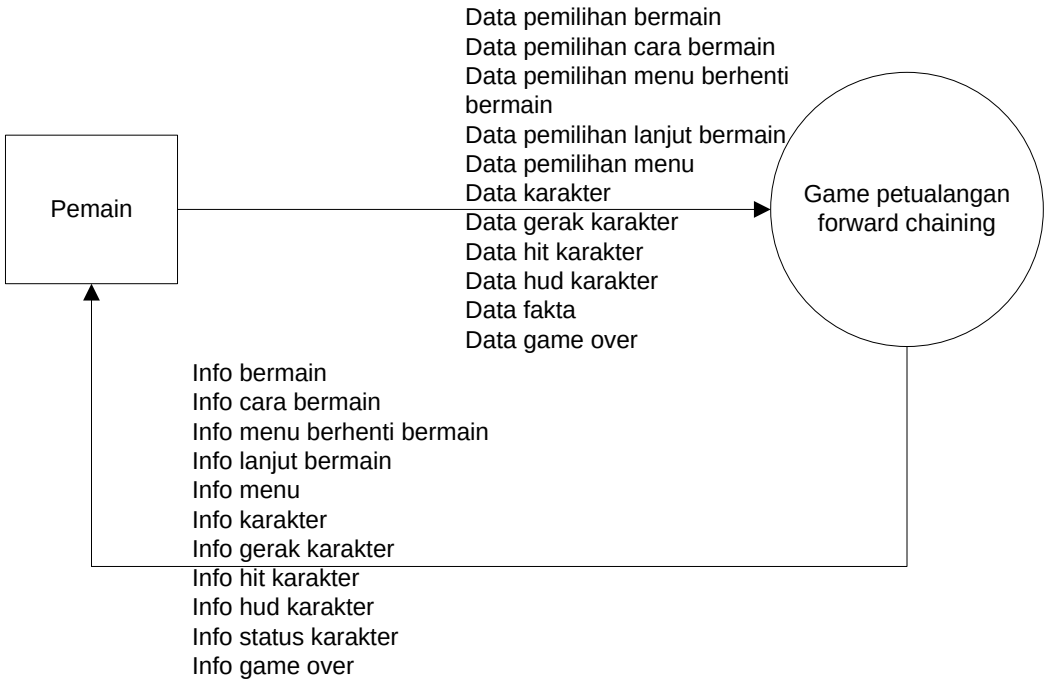
Peran	Pemain
Usia	12 tahun ke atas
Tingkat Keterampilan	Dapat menggunakan komputer
Kondisi Fisik	Tidak buta total dan tidak buta warna.

3.7. Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional menggambarkan proses-proses yang diterapkan, agar game dapat berjalan dengan baik.

3.7.1. Diagram Konteks

Diagram konteks adalah diagram yang menggambarkan *input*, proses dan *output* pada sistem perangkat lunak yang akan dibangun, berikut diagram konteks dari *game* petualangan dapat dilahat pada Gambar 3.9.



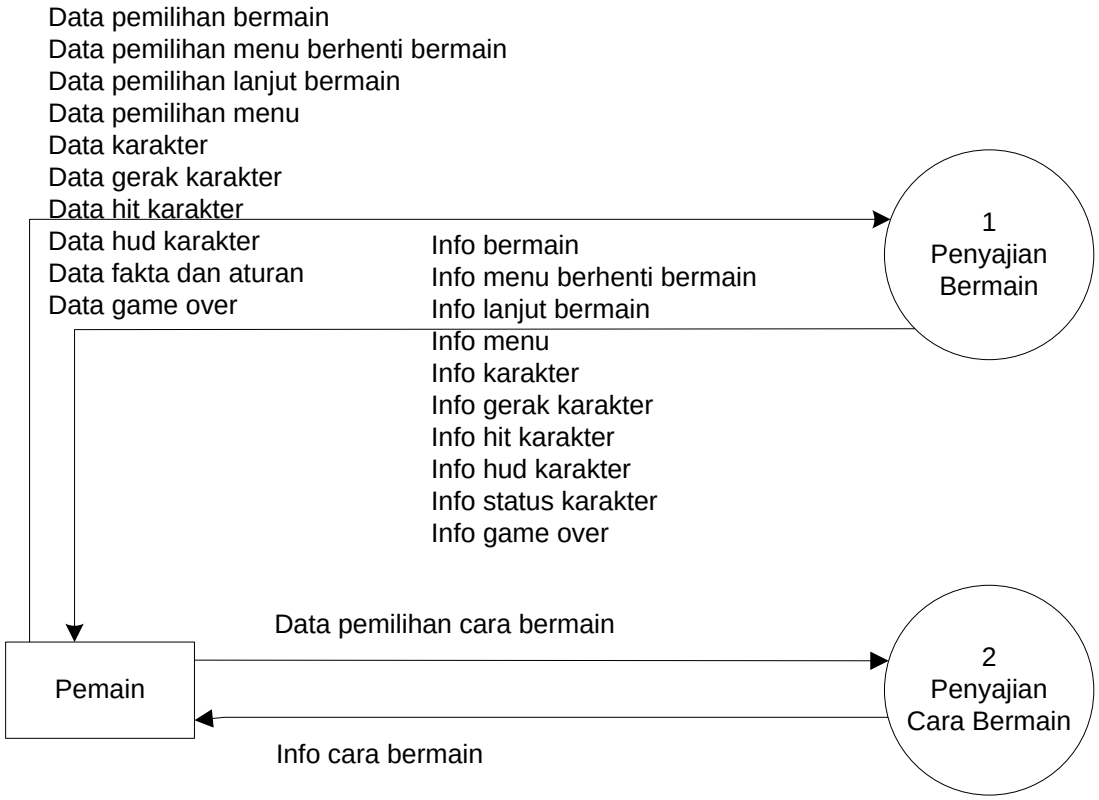
Gambar 3.9 Diagram Konteks *Game* Petualangan *Forward Chaining*

3.7.2. *Data Flow Diagram*

Data Flow Diagram (DFD) adalah suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa saja yang menghasilkan data tersebut dan interaksi antara data yang disimpan dan proses yang dikenakan pada data tersebut.

3.7.2.1. DFD Level 1

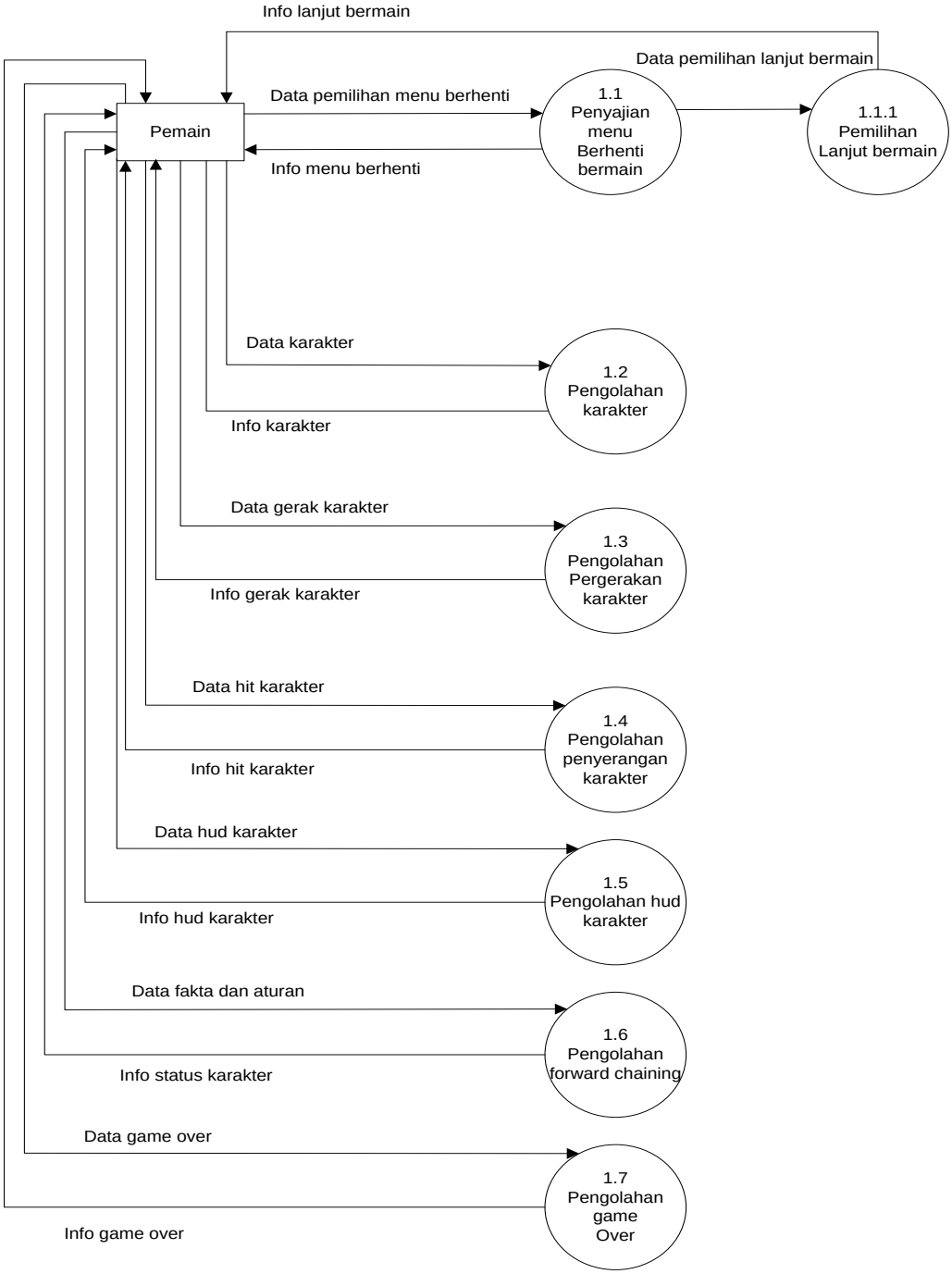
Data Flow Diagram level 1 mempunyai 3 proses, diantaranya adalah *play*, *how to play*, dan *exit*, berikut DFD dapat dilihat pada Gambar 3.10.



Gambar 3.10 DFD Level 1

3.7.2.2. DFD Level 2 Proses 1

Berikut DFD Level 2 Proses 1 dapat dilihat pada Gambar 3.11.



Gambar 3.11 DFD Level 2 Proses 1

3.7.2.3. Spesifikasi Proses

Spesifikasi proses digunakan untuk menggambarkan model aliran yang terdapat pada DFD. Spesifikasi proses DFD akan dijelaskan pada Tabel 3.17.

Tabel 3.17 Spesifikasi Proses

No	Proses	Keterangan
1	No. Proses	1.1
	Nama Proses	Penyajian menu berhenti bermain
	Deskripsi	Proses untuk menampilkan menu berhenti bermain
	Source	Pemain
	Input	Data pemiliham menu berhenti bermain
	Output	Info menu berhenti bermain
	Destination	Pemain
	Logika Proses	1. Pemain memasukan data pemilihan menu berhenti bermain 2. Sistem akan menampilkan informasi menu berhenti bermain
2	No. Proses	1.1.1
	Nama Proses	Pemilihan lanjut bermain
	Deskripsi	Proses untuk meneruskan game
	Source	Penyajian menu berhenti bermain
	Input	Data pemilihan lanjut bermain
	Output	Info lanjut bermain
	Destination	Pemain
	Logika Proses	1. Pemain memasukan data pemilihan lanjut bermain 2. Sistem akan menampilkan informasi lanjut bermain
3	No. Proses	1.2
	Nama Proses	Pengolahan karakter
	Deskripsi	Proses untuk menampilkan karakter
	Source	Pemain
	Input	Data karakter
	Output	Info karakter
	Destination	Pemain
	Logika Proses	1. Pemain memasukan data karakter 2. Sistem akan menampilkan informasi karakter

Tabel 3.17 Spesifikasi Proses (Lanjutan)

No	Proses	Keterangan
4	No. Proses	1.3
	Nama Proses	Pengolahan pergerakan karakter
	Deskripsi	Proses untuk menggerakan karakter
	Source	Pemain
	Input	Data gerak karakter
	Output	Info gerak karakter
	Destination	Pemain
	Logika Proses	1. Pemain memasukan data gerak karakter 2. Sistem akan menampilkan informasi gerak karakter
5	No. Proses	1.4
	Nama Proses	Pengoalahan penyerangan karakter
	Deskripsi	Proses untuk mengeluarkan serangan karakter
	Source	Pemain
	Input	Data hit karakter
	Output	Info hit karakter
	Destination	Pemain
	Logika Proses	1. Pemain memasukan data hit karakter 2. Sistem akan menampilkan informasi hit karakter
6	No. Proses	1.5
	Nama Proses	Pengoalahan hud karakter
	Deskripsi	Proses untuk menampilkan hud
	Source	Pemain
	Input	Data hud karakter
	Output	Info hud karakter
	Destination	Pemain
	Logika Proses	1. Pemain memasukan data hud karakter 2. Sistem akan menampilkan informasi hud karakter

Tabel 3.17 Spesifikasi Proses (Lanjutan)

No	Proses	Keterangan
7	No. Proses	1.6
	Nama Proses	Pengolahan forward chaining
	Deskripsi	Proses untuk pengecekan fakta sesuai aturan
	Source	Pemain
	Input	Data fakta dan aturan
	Output	Info status karakter
	Destination	Pemain
	Logika Proses	1. Pemain memasukan data fakta sesuai aturan 2. Sistem akan menapilkan informasi status karakter
8	No. Proses	1.7
	Nama Proses	Pengolahan game over
	Deskripsi	Proses untuk menampilkan game over
	Source	Pemain
	Input	Data game over
	Output	Info game over
	Destination	Pemain
	Logika Proses	1. Pemain memasukan data game over 2. Sistem akan menampilkan informasi game over

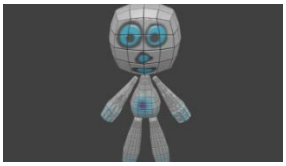
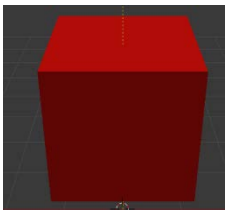
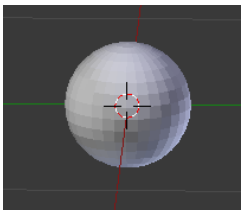
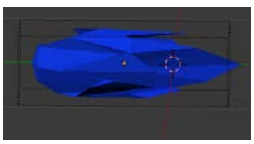
3.7.3. Perancangan *Game*

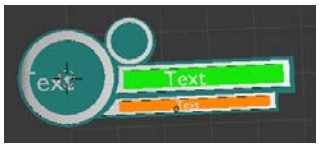
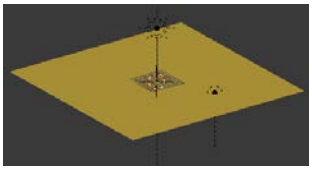
Perancangan *game* menjelaskan mengenai perancangan konten-konten game, perancangan struktur menu, perancangan antarmuka, perancangan jaringan semantik, dan perancangan prosedural.

3.7.3.1. Perancangan Konten *Game*

Perancangan konten *game* menjelaskan mengenai konten-konten yang berada pada *game*, berikut konten-konten *game* dapat dilihat pada Tabel 3.18.

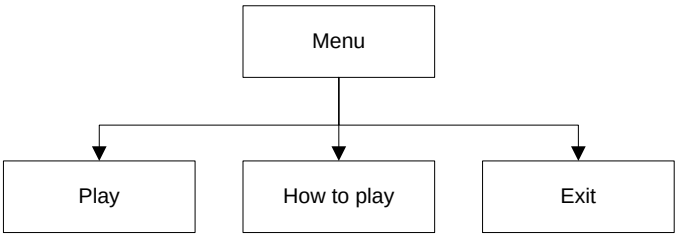
Tabel 3.18 Konten-konten Game

No.	Nama	Konten	Keterangan
1	Karakter		1.Karakter utama yang dimainkan oleh player 2.Dapat berjalan,mundur,belok kiri,belok kanan,melompat,lari,menyerang musuh.
2	Musuh		1.Musuh yang harus dihadapi karakter 2.Dapat menyerang karakter
3	Boll		Skill yang dikeluarkan karakter
4	Fireboll		Skill yang dikeluarkan pada karakter level 1
5	Iceboll		Skill yang dikeluarkan pada karakter level 2

No.	Nama	Konten	Keterangan
6	Hud		Keterangan status karakter
7	Pohon 1		Properti biasa yang ada pada game
8	Pohon 2		Properti biasa yang ada pada game
9	Ground		Properti ground sebagai tempat berpijak karakter

3.7.3.2. Perancangan Struktur Menu

Perancangan struktur menu berisikan menu dan submenu yang berfungsi memudahkan *user* didalam menggunakan aplikasi. Untuk lebih jelasnya dapat dilihat pada Gambar 3.12.



Gambar 3.12 Struktur Menu *Game*

Berikut penjelasan struktur menu permainan.

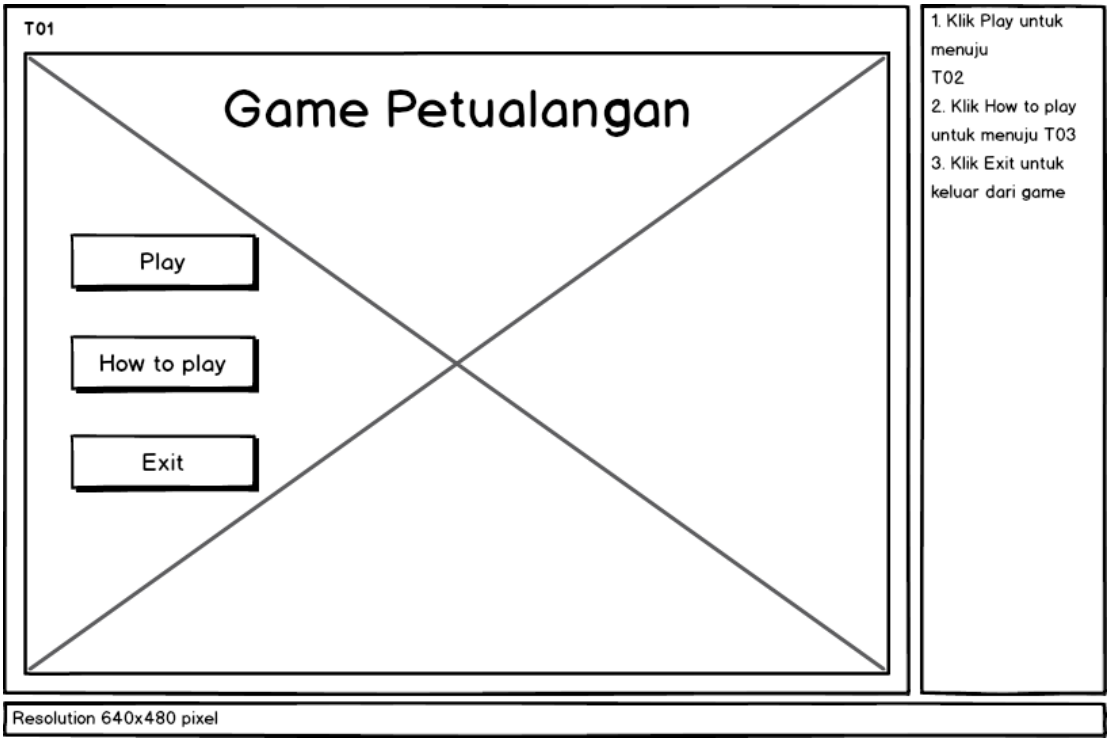
1. Menu Play : Menu untuk memulai *csene game.*
2. Menu How to play : Menu untuk menampilkan *scene to play.*
3. Menu Exit : Menu untuk keluar dari *game*.

3.7.3.3. Perancan Antarmuka

Perancangan antarmuka bertujuan untuk memberikan gambaran tentang *game* yang akan dibangun, sehingga akan mempermudah dalam implementasi dan pembuatan *game*. Berikut perancangan antar muka *game* petualangan dapat dilihat pada Gambar 3.13 sampai Gambar 3.18.

1. Perancangan Antarmuka Menu Utama

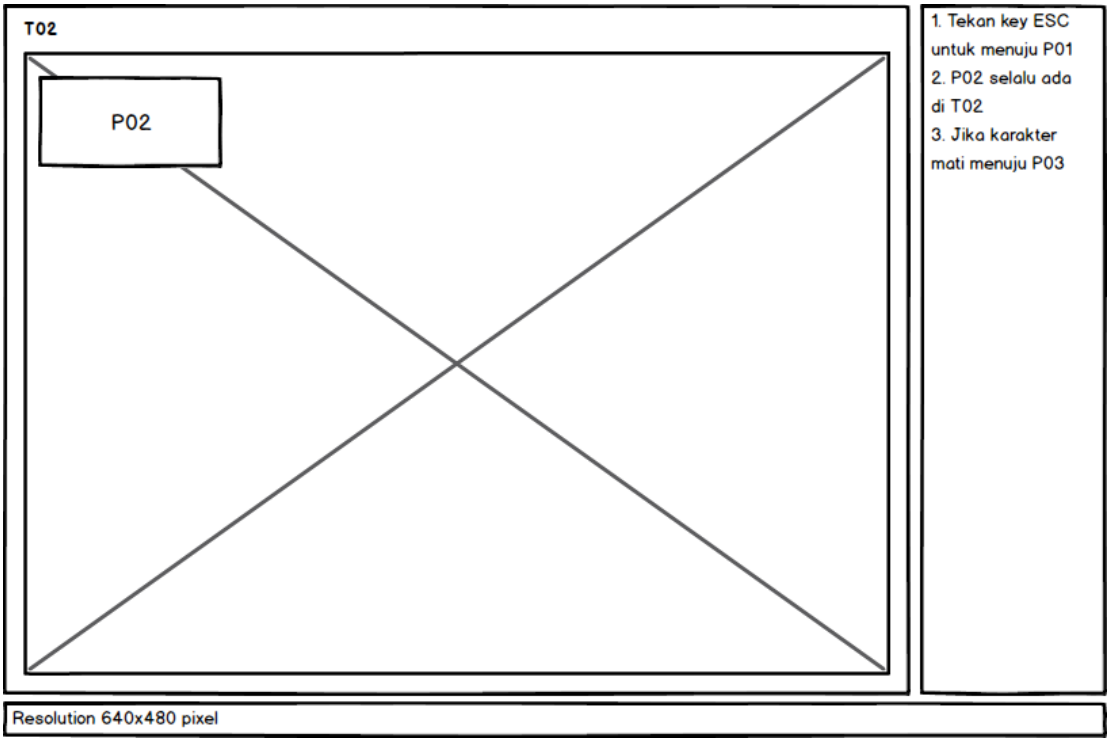
Perancangan antarmuka menu utama merupakan konsep atau gambaran menu utama pada *game* nantinya, pada menu utama terdapat tombol *play* untuk memulai permainan, tombol *how to play* untuk melihat cara bermain, dan tombol *exit* untuk keluar dari *game*.



Gambar 3.13 Perancangan Antarmuka Menu Utama

2. Perancangan Antarmuka Game

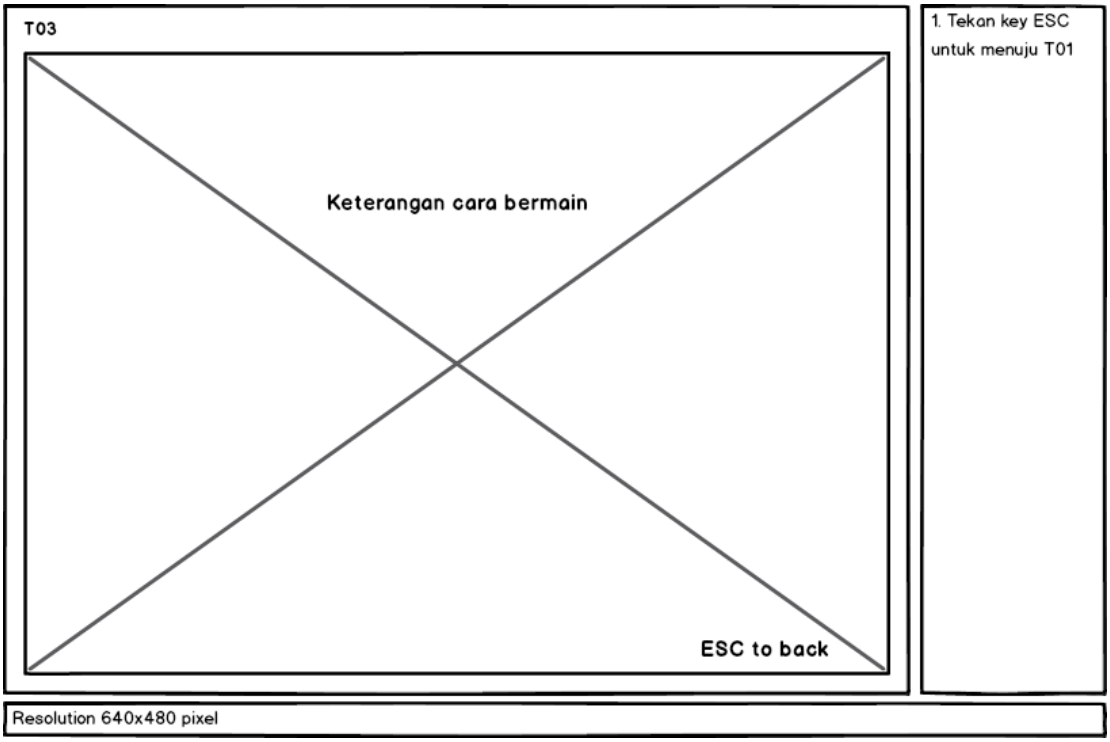
Perancangan antarmuka game merupakan konsep atau gambaran *game* pada *game* nantinya, pada perancangan antarmuka game terdapat HUD yang selalu tampil pada *game*.



Gambar 3.14 Perancangan Antarmuka Game

3. Perancangan Antarmuka Menu How to Play

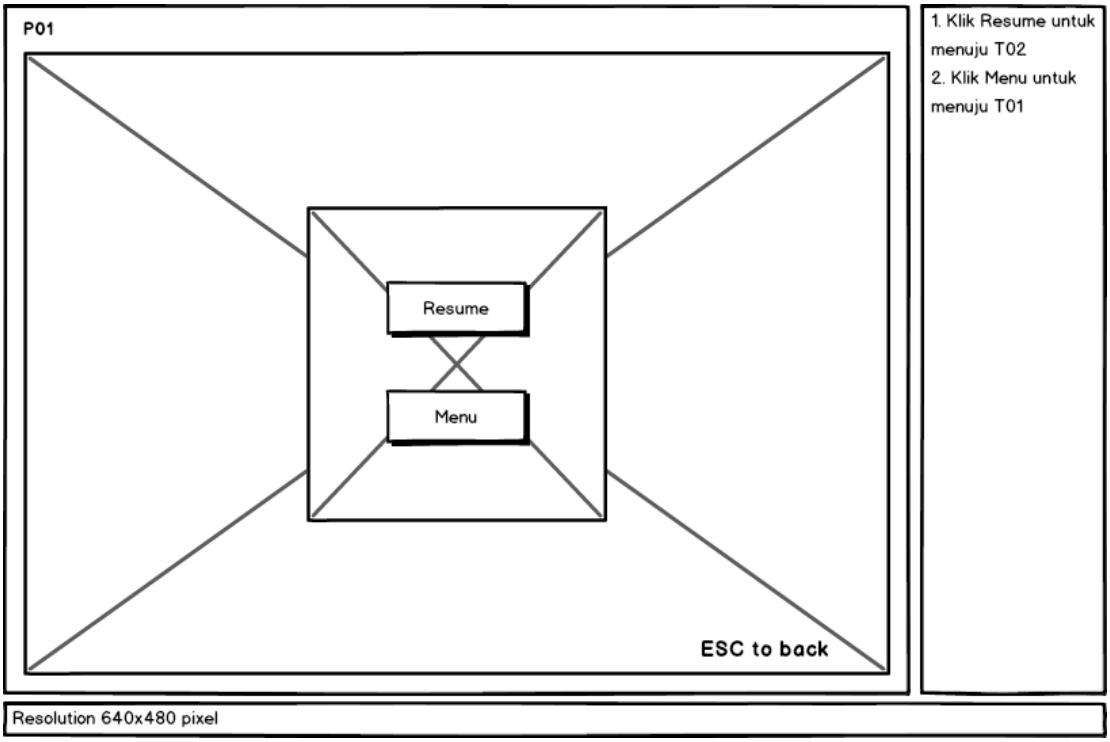
Perancangan antarmuka *menu how to play* merupakan konsep atau gambaran keterangan cara bermain pada *game* nantinya, pada perancangan *how to play* terdapat keterangan cara bermain *game*.



Gambar 3.15 Perancangan Antarmuka How to Play

4. Perancangan Antarmuka Pesan Pause

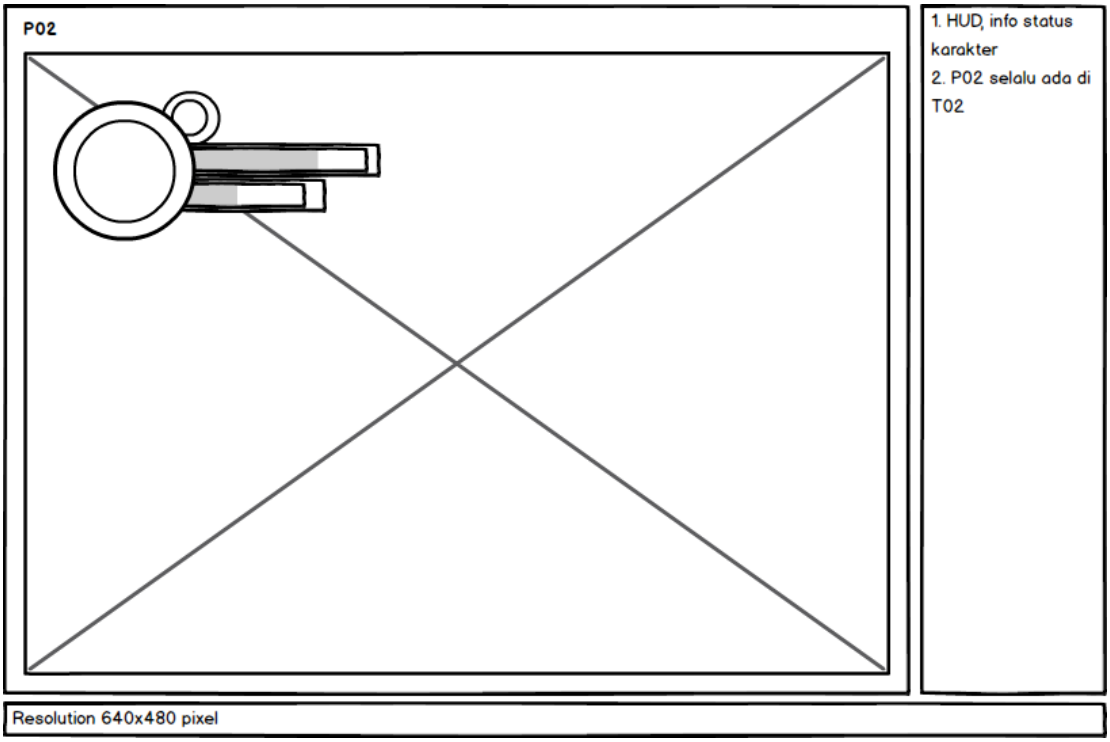
Perancangan antarmuka pesan *pause* merupakan konsep atau gambaran pesan *pause* pada *game* nantinya, pada perancangan pesan *pause* terdapat tombol *resume* untuk meneruskan permainan, dan tombol menu untuk kembali ke menu utama.



Gambar 3.16 Perancangan Antarmuka Pause

5. Perancangan Antarmuka HUD

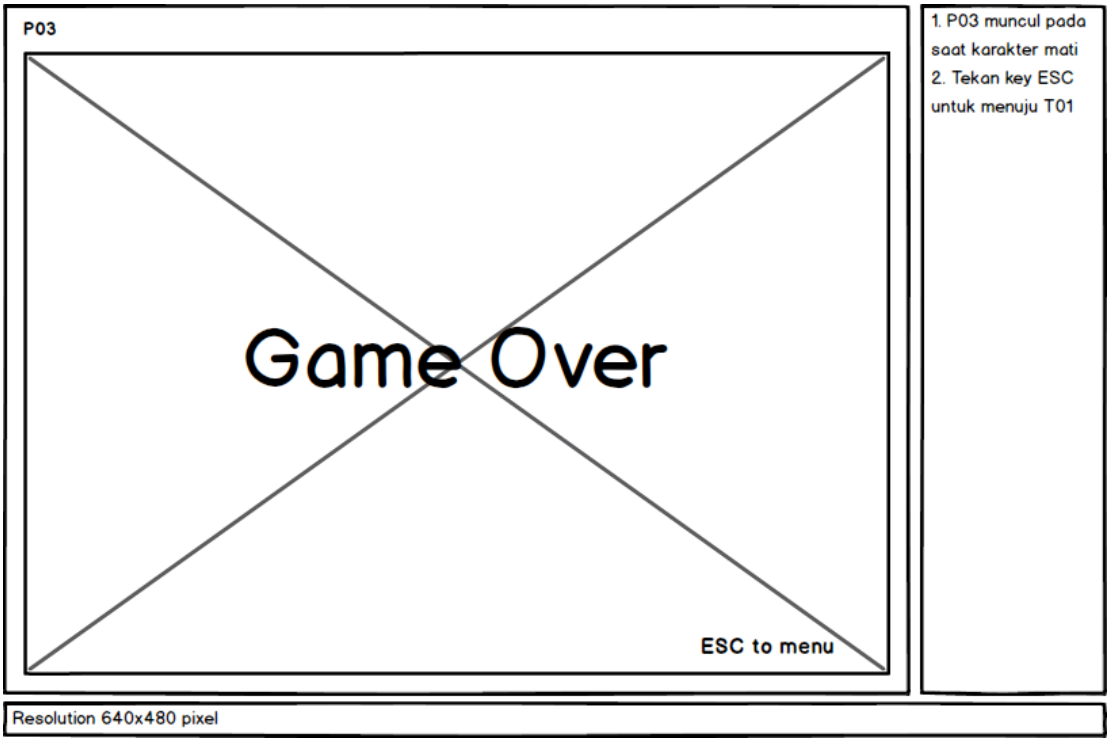
Perancangan antarmuka HUD (*Head Up Display*) merupakan konsep atau gambaran HUD pada *game* nantinya, pada perancangan antarmuka HUD ini terdapat info-info karakter, berupa hp, exp dan level.



Gambar 3.17 Perancangan Antarmuka HUD

6. Perancangan Antarmuka Pesan Game Over

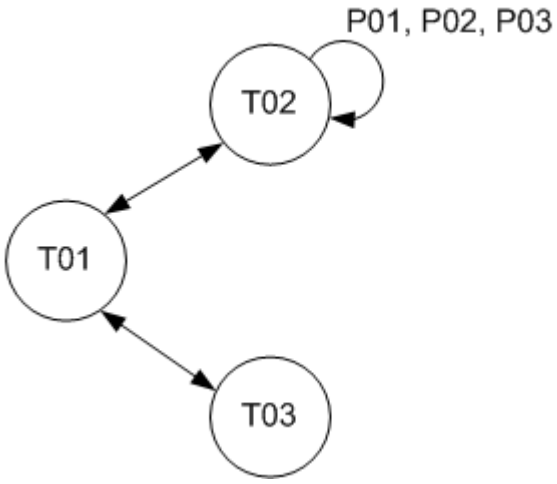
Perancangan antarmuka pesan *game over* merupakan konsep atau gambaran dari pesan *game over* pada *game* nantinya, pada perancangan antarmuka pesan *game over* ini terlihat keterangan *game over*, pesan ini muncul pada saat karakter mati.



Gambar 3.18 Perancangan Antarmuka Game Over

3.7.3.4. Jaringan Semantik

Jaringan semantik digunakan untuk menghubungkan setiap tampilan (interface) pada sebuah aplikasi yang dirancang, untuk lebih jelasnya dapat dilihat pada Gambar 3.19.



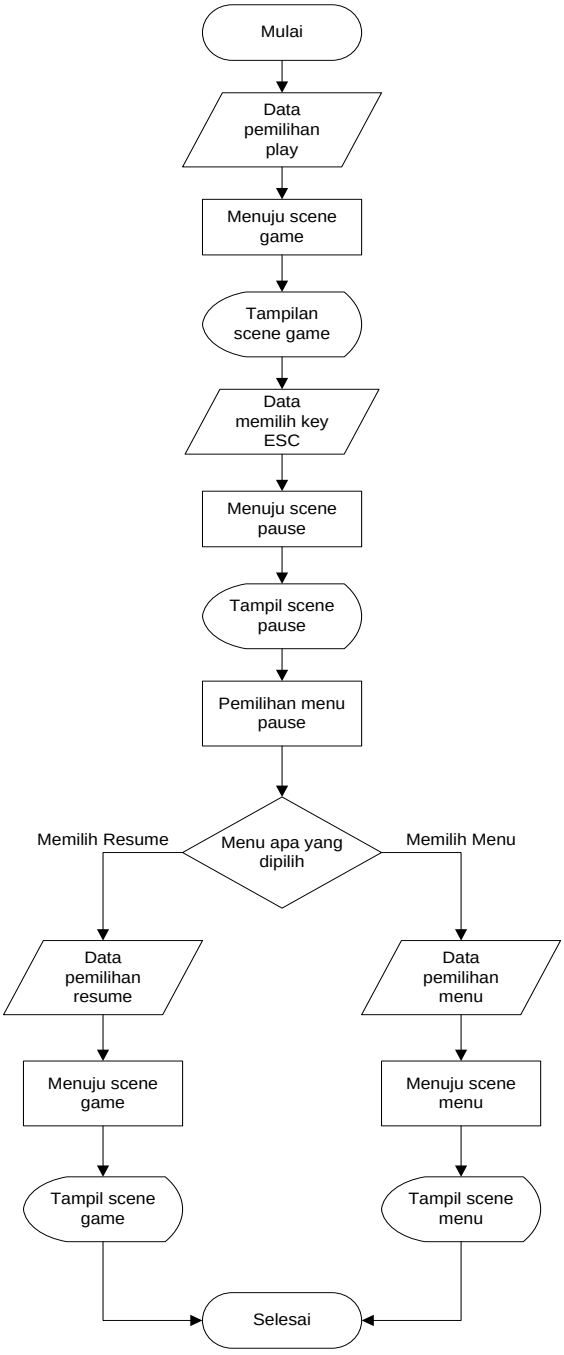
Gambar 3.19 Jaringan Semantik

3.7.3.5. Perancangan Prosedural

Sebagai langkah terakhir dalam perancangan yaitu perancangan prosedural yang akan diimplementasikan ke dalam sistem. Prosedural ini akan digunakan sebagai algoritma dasar dalam mengkodekan prosedur yang ada. Perancangan prosedural untuk aplikasi yang akan dibangun dapat dilihat pada Gambar 3.20 sampai Gambar 3.24.

1. Perancangan Prosedural *Play*

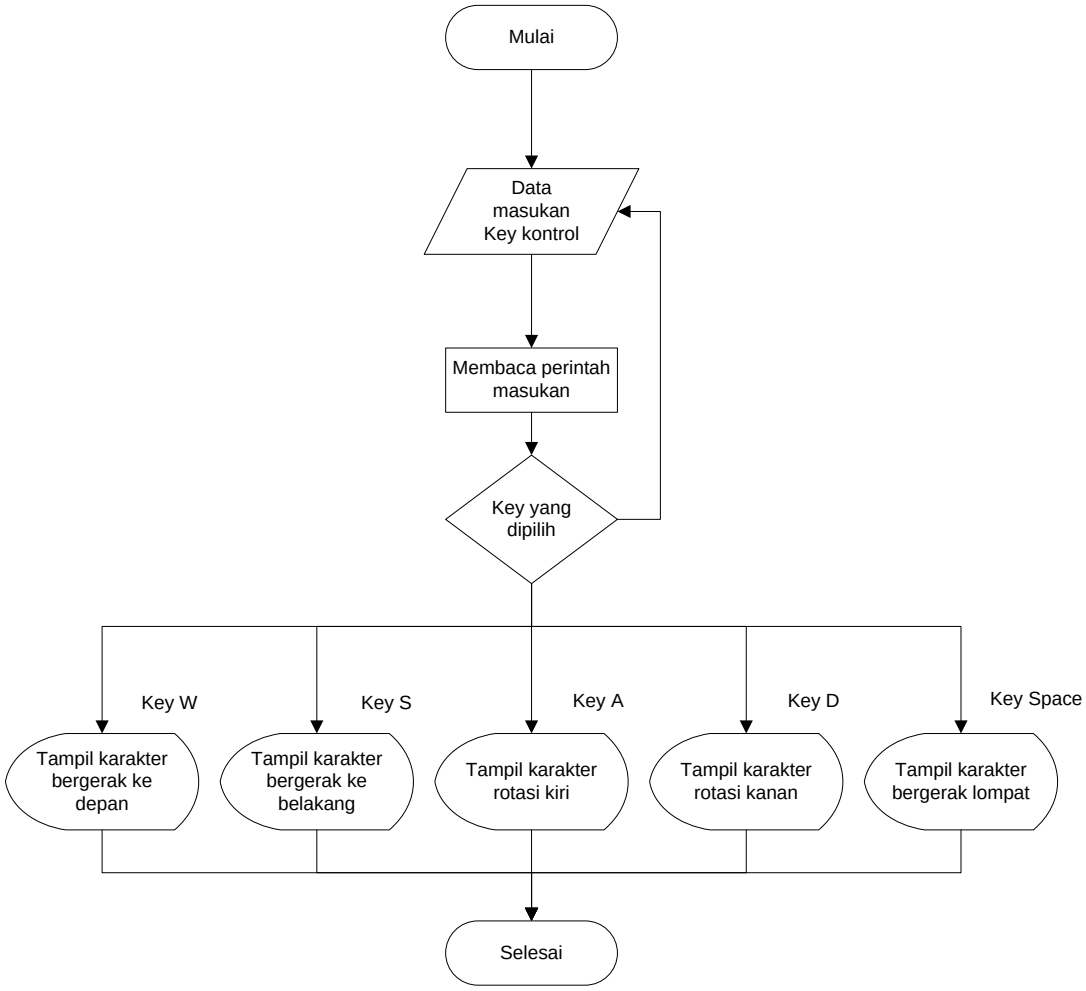
Perancangan prosedural *play* merupakan gambaran proses memulai permainan.



Gambar 3.20 *Flowchart Play*

2. Perancangan Prosedural Gerak Karakter

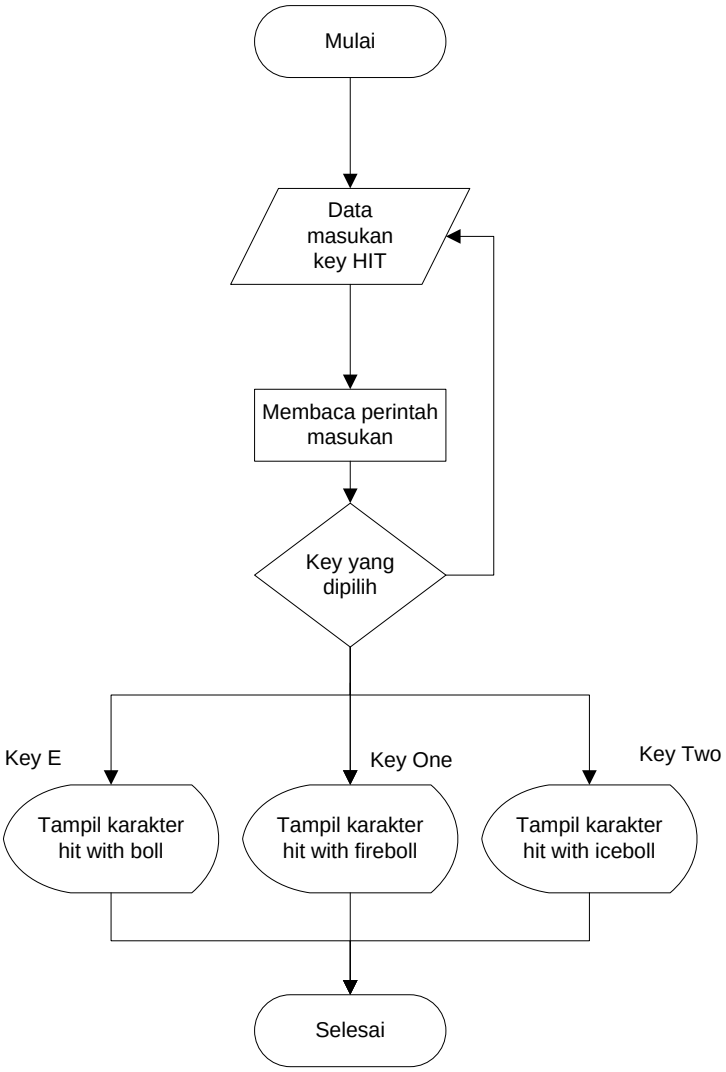
Perancangan prosedural gerak karakter merupakan gambaran proses gerak karakter pada *game*.



Gambar 3.21 *Flowchart* Gerak Karakter pada Proses Pergerakan Karakter

3. Perancangan Prosedural Karakter Hit

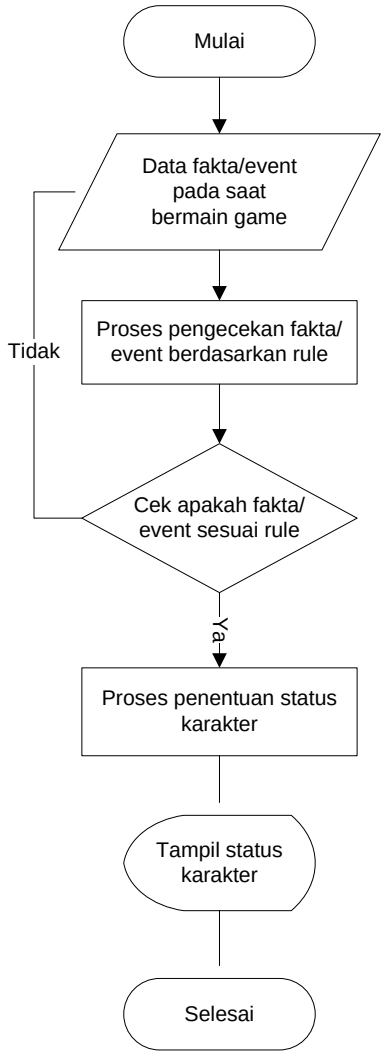
Perancangan prosedural karakter hit merupakan gambaran proses karakter hit pada *game.*



Gambar 3.22 *Flowchart* Karakter Hit pada Proses Penyerangan Karakter

4. Perancangan Prosedural *Forward Chaining*

Perancangan prosedural *forward chaining* merupakan gambaran proses *forward chaining* pada *game.*



Gambar 3.23 *Flowchart Forward Chaining*